APPLICATION PROTOCOLS

IT SECURITY ACADEMY
www.SecAcademy.com

PRESENTATION LAYER

Threats: Incorrect Data Redndering

The presentation layer's function is to code and decode transmitted data packets. The attacks you may face here relate to attackers potentially manipulating how the data is rendered in web programs

SEVERAL YEARS AGO AN IIS SERVER VULNERABILITY WAS DISCOVERED THAT ENABLED ATTACKERS TO:

USE

HTTP to access a targeted computer's hard disk running an IIS server. The URI address needed to be submitted in hexadecimals

DOWNLOAD

files from an IIS server's hard disk by executing the GET command with the file name expressed in Unicode

PRESENTATION LAYER

Threats: Incorrect Data Redndering

FOR BOTH INSTANCES standard character requests were denied by the server. IIS checked if they contain the ../. string that allows a client to move up to a parent folder

BUT IF THE ATTACKER decided to use Unicode to code the / character as %c0%af, the IIS server failed to detect the string, allowing the user to move up to a parent folder of his choosing

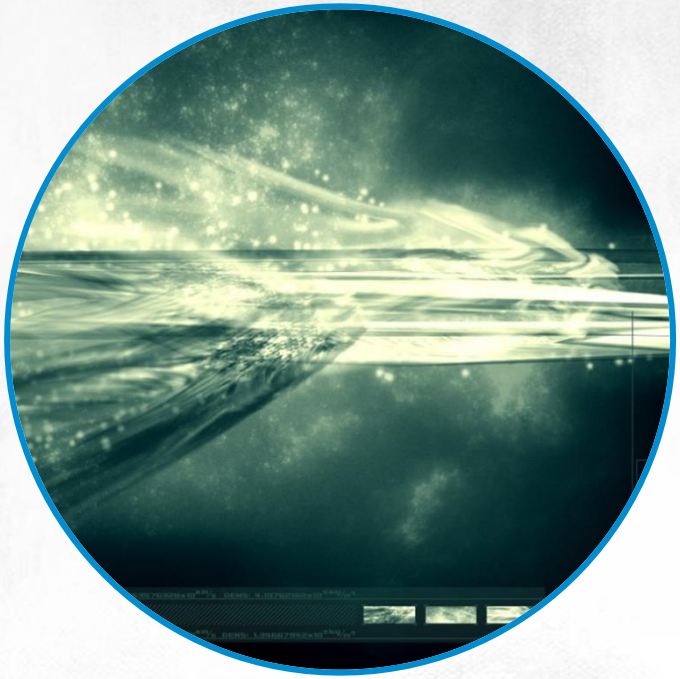
THE UNDERLYING PROBLEM was that IIS servers accept and execute commands coded using a code page and Unicode, but only the standard, normal characters were checked against a black list of forbidden requests

PRESENTATION LAYER

Threats: Null Byte Injection

THIS ATTACK THREAT IS NOT EXCLUSIVE TO SERVERS AND IS MORE PREVALENT IN PROGRAMMING LANGUAGES

Null Byte Injection as a vulnerability found in most web applications that use high-level languages like PHP, ASP.Net, Perl or Java. These applications still have to communicate with web servers or operating systems, the majority of which are written in C or C++. Both of them have Null represent the end of a string character. In other words, everything that comes after this character will not be rendered



PRESENTATION LAYER

Threats: Null Byte Injection

IF AN APPLICATION IS WRITTEN IN PHP AND USES THE FOLLOWING FUNCTION TO RETRIEVE THE .DAT FILE:

```
$file = $_GET['file'];
require_once("/var/www/images/$file.dat");
```

THE ATTACKER IS ABLE TO DOWNLOAD ANY FILE STORED IN THE WEB SERVER JUST BY CHANGING THE STANDARD REQUEST

http://www.server.com/user.php?file=myprofile.dat

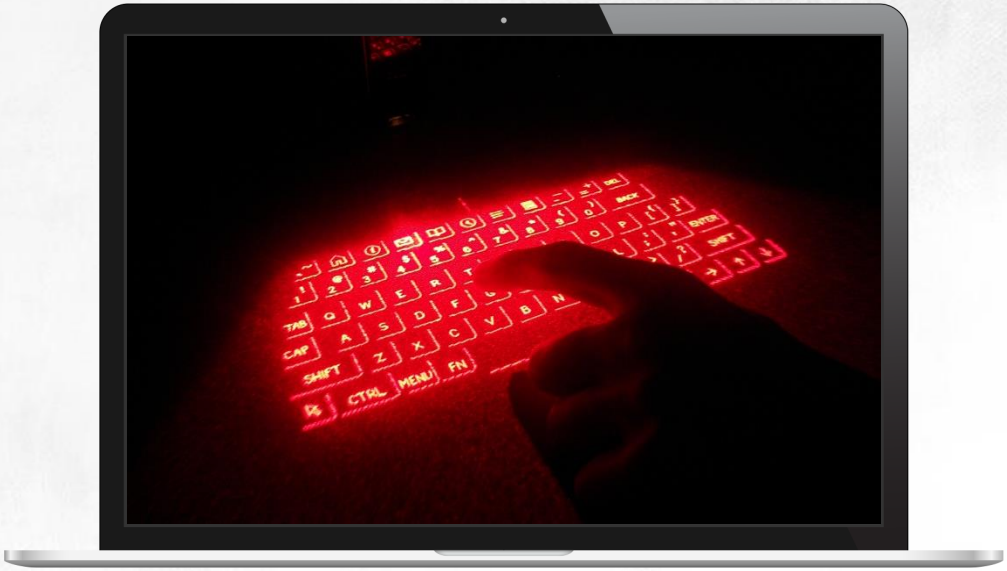
into this one:
http://www. server.com /user.php?file=../../../etc/passwd%00

PRESENTATION LAYER

Threats: Null Byte Injection

IF AN APPLICATION THAT IS WRITTEN IN JAVA USES THIS FUNCTION TO LOAD A FILE:

```
String fn =
request.getParameter("fn");
if (fn.endsWith(".db"))
{
File f = new File(fn);
//loading content of file f
...
```



PRESENTATION LAYER

Threats: Null Byte Injection

THIS FUNCTION SHOULD ONLY ENABLE YOU TO LOAD FILES WITH A SPECIFIC EXTENSION

BUT if the attacker changes this request:
http:// www.server.com/mypage.jsp?fn=report.db

INTO A REQUEST where the correct file name is preceded with Null, he is going to be able to load the content of any file:
http:// www.server.com /mypage.jsp?fn=serverlogs.txt%00.db

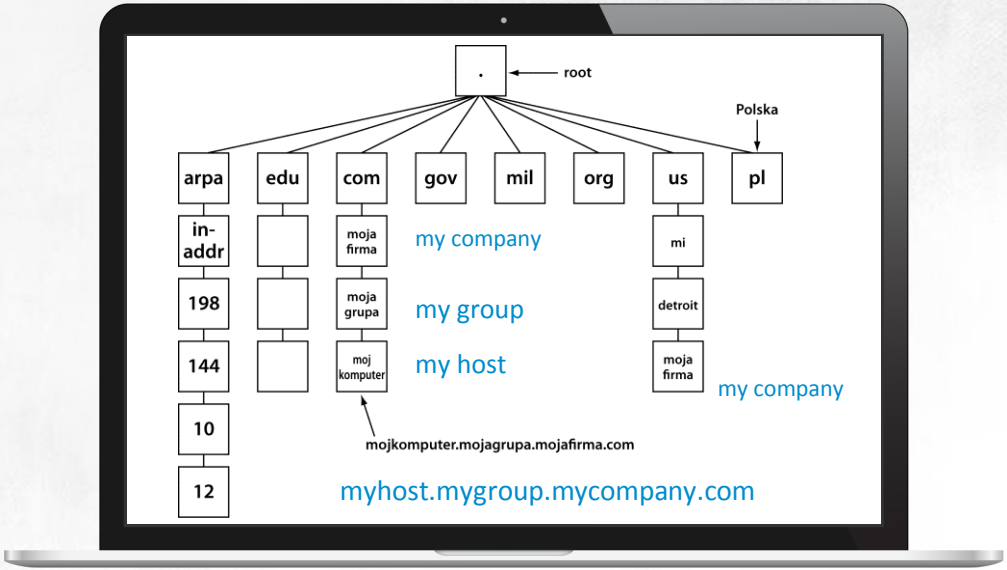
IT IS WEB APPLICATION DEVELOPERS WHO ARE ACCOUNTABLE FOR SECURING THEIR PRODUCTS AGAINST THESE THREATS

APPLICATION LAYER

Threats: Confidential Information Disclosure

The DNS constitutes one of the most fundamental Internet services, and consists of two parts:

- **A DNS SERVER,** which responds to client requests
- **A DNS CLIENT (RESOLVER),** which is part of the OS



APPLICATION LAYER

Threats: Confidential Information Disclosure

THE DNS is built around a hierarchical stricture

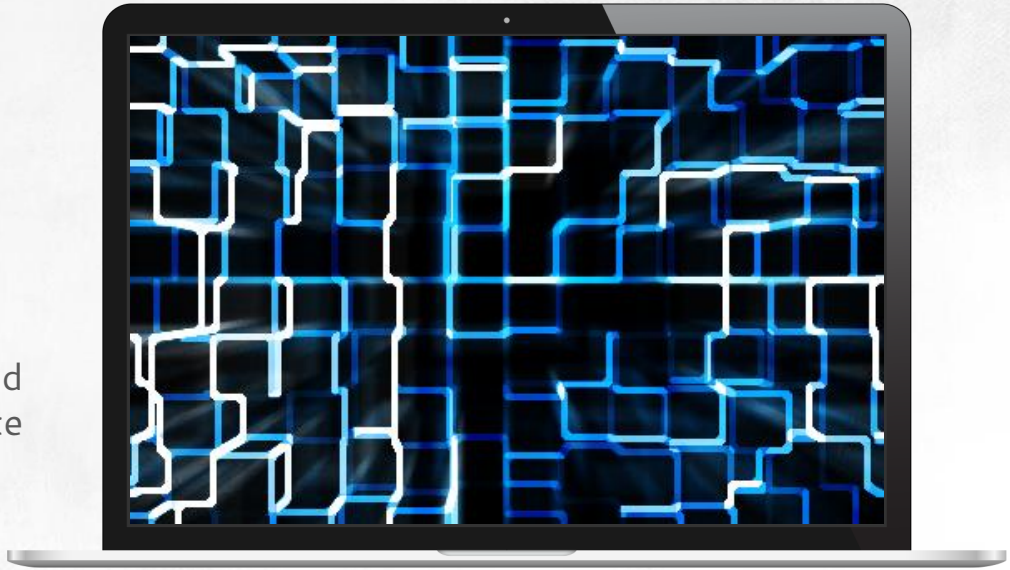
DNS SERVERS STORE information about hosts (info stored as A record types) and their canonical names (as CNAME records), as well as info about some network services like DNS servers (as NS records) and mail servers (as MX records)

EXERCISE

Application Layer Attack

READING
registry data:
http://serversniff.net

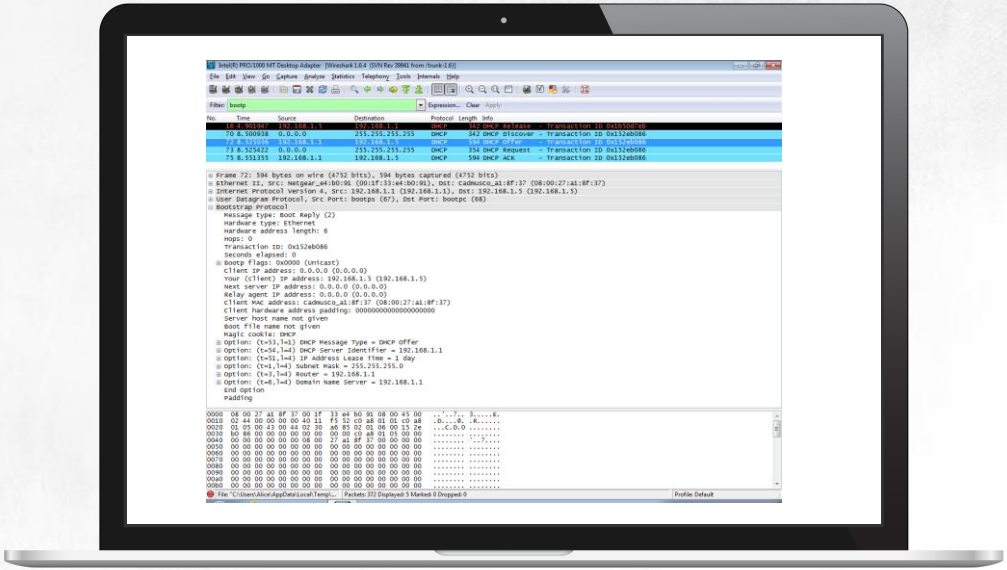
READING
registry data using dig and the dnsrecon and fierce scripts



APPLICATION LAYER

Threats: Man-in-the-Middle and Denial od Service Attacks

THE DHCP PROTOCOL is widely used in nearly all networks



APPLICATION LAYER

Threats: Man-in-the-Middle and Denial od Service Attacks

This is how it works:

A client computer broadcasts a DHCP discover message to locate the servers in a network (DHCPDISCOVER)

The DHCP servers then replies with an offer to assign an IP address to the client making the request (DHCPOFFER)

Then, the client broadcasts a request to get the IP address (DHCPREQUEST)

One of the available DHCP servers confirms the offer and IP assignment (the DHCPACK message) or informs the client the request has been denied (DHCPNAK)

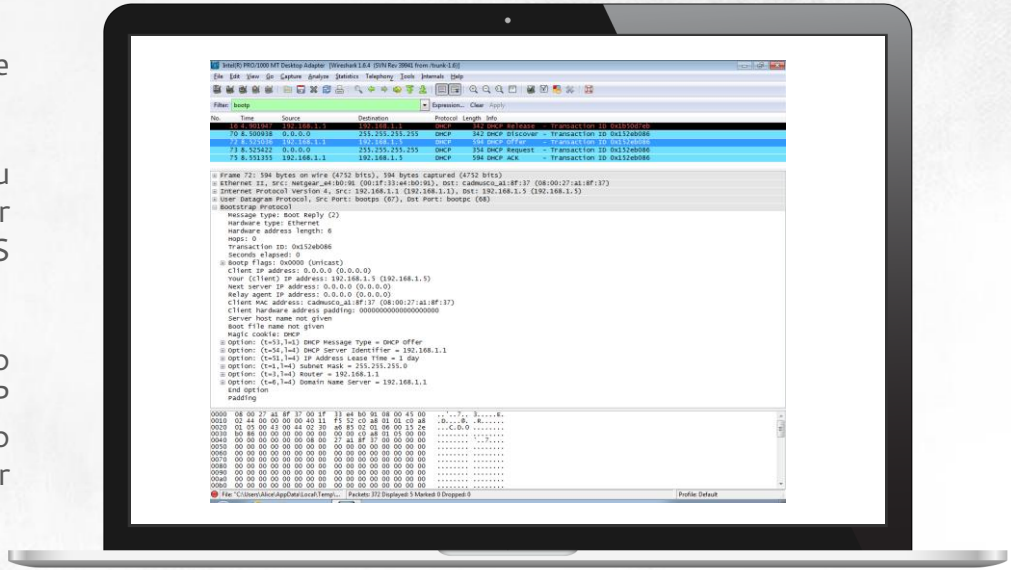
APPLICATION LAYER

Threats: Man-in-the-Middle and Denial od Service Attacks

THE CLIENT makes no attempt to verify the identity of the DHCP server

IF YOU RUN your own DHCP server, you can direct traffic through your own router or make client computers use your DNS server

DOS attacks are equally straightforward to launch: it's enough to connect a DHCP server to a network and configure it to assign clients the IP addresses of other networks



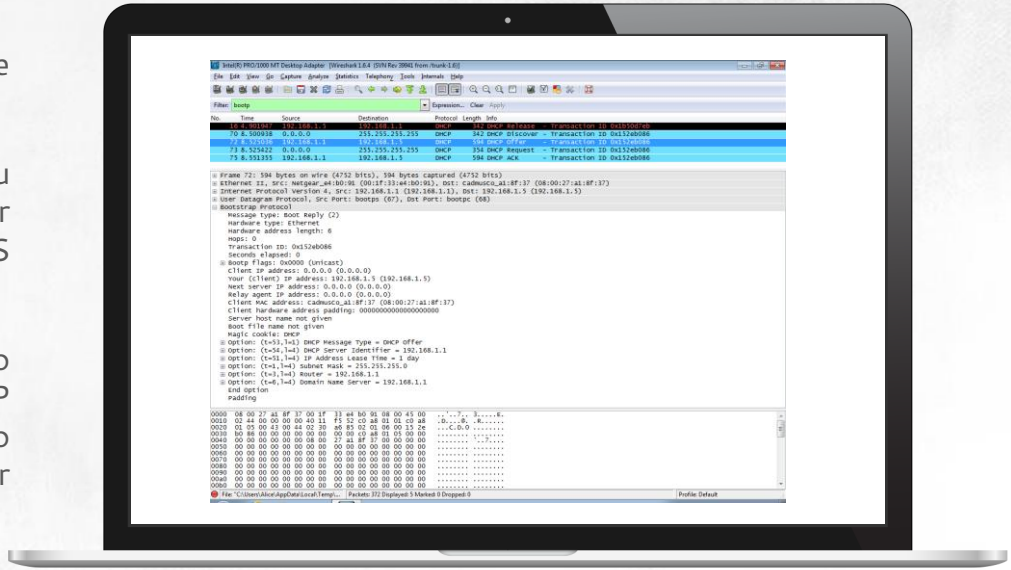
APPLICATION LAYER

Threats: Man-in-the-Middle and Denial od Service Attacks

THE CLIENT makes no attempt to verify the identity of the DHCP server

IF YOU RUN your own DHCP server, you can direct traffic through your own router or make client computers use your DNS server

DOS attacks are equally straightforward to launch: it's enough to connect a DHCP server to a network and configure it to assign clients the IP addresses of other networks



APPLICATION LAYER

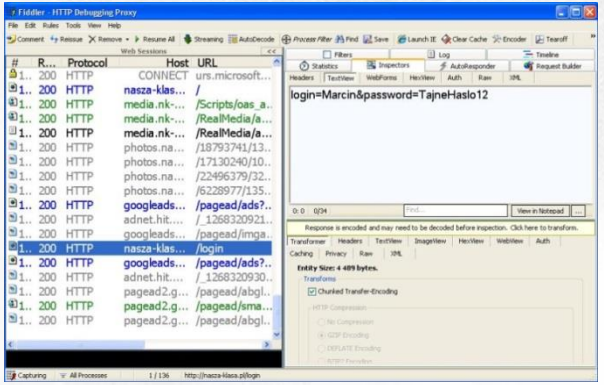
Threats: Eavesdropping on and Modifying Transmitted Packets

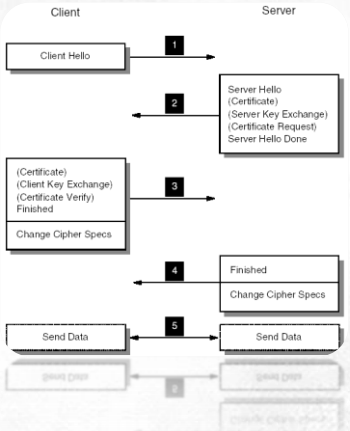
HTTP TRANSMITS all data packets in the plaintext format. The reason for this is that most Internet resources are publically available, and thus there is no need to encrypt them

HTTP also runs the risk of on-the-fly modification of data. The attacker can:

- Modify data sent to web servers
- Modify files downloaded from web servers. Even if the program you want to download is itself trusted and the website is also secure, attackers can still append malware to it

HTTPS aims to provide the confidentiality and ensure the authenticity of transmitted packets, but what about Man-in-the-Middle attacks?





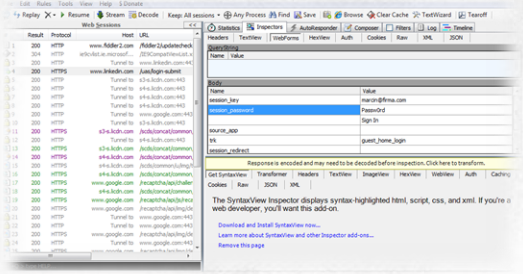
APPLICATION LAYER

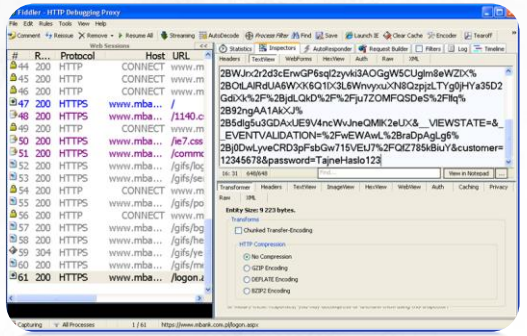
Threats: Eavesdropping on and Modifying Transmitted Packets

IF THE CLIENT DOESN'T CONNECT to a web server directly and uses the attacker's computer for that purpose, while the web application only uses https to give data a layer of protection, the attacker will obtain full access to confidential data...

...Provided the user ignores a web browser warning about the invalid certificate

IF THAT'S THE CASE, the user will send to the attacker a password encrypted using a certificate that the attacker has, and the attacker will decrypt it and then encrypt it again using the web server's certificate and send it back to the https address entered by the user

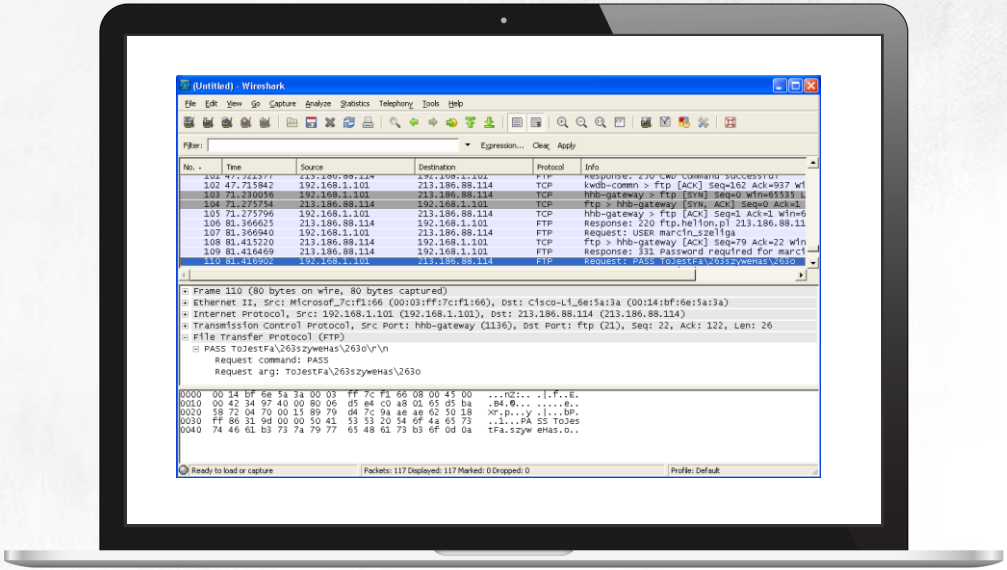




APPLICATION LAYER

Threats: Eavesdropping on Transmitted Packets and Password Cracking

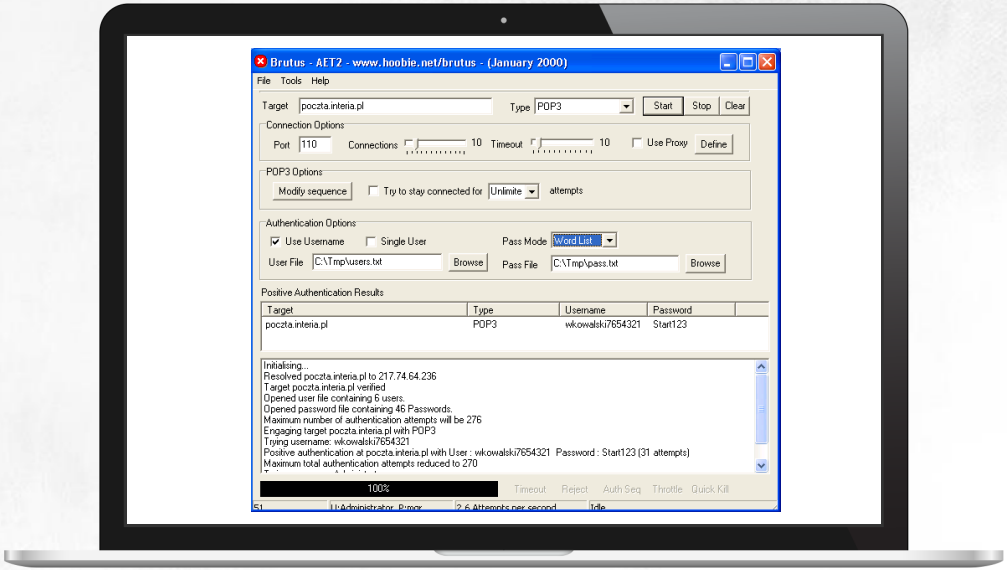
ALL DATA TRANSMITTED
using the FTP protocol, including credentials, are in the plaintext format


(Untitled) - Wireshark
File Edit View Go Capture Analyze Statistics Telephony Tools Help
Filter:
Expression... Clear Apply
No. Time Source Destination Protocol Info
102 47.715842 192.168.1.101 213.186.88.114 TCP kwdb-commn > ftp [ACK] Seq=162 Ack=937 Wi
103 71.230056 192.168.1.101 213.186.88.114 TCP hhb-gateway > ftp [SYN] Seq=0 Win=65535 L
104 71.275754 213.186.88.114 192.168.1.101 TCP ftp > hhb-gateway [SYN, ACK] Seq=0 Ack=1
105 71.275796 192.168.1.101 213.186.88.114 TCP hhb-gateway > ftp [ACK] Seq=1 Ack=1 Win=6
106 81.366625 213.186.88.114 192.168.1.101 FTP Response: 220 ftp.hellon.pl 213.186.88.11
107 81.366940 192.168.1.101 213.186.88.114 FTP Request: USER marcin_szeliga
108 81.415220 213.186.88.114 192.168.1.101 TCP ftp > hhb-gateway [ACK] Seq=79 Ack=22 Win
109 81.416469 213.186.88.114 192.168.1.101 FTP Response: 331 Password required for marci
110 81.416902 192.168.1.101 213.186.88.114 FTP Request: PASS ToJestFa\263szyweHas\263o
Frame 110 (80 bytes on wire, 80 bytes captured)
Ethernet II, Src: Microsof_7c:f1:66 (00:03:ff:7c:f1:66), Dst: Cisco-Li_6e:5a:3a (00:14:bf:6e:5a:3a)
Internet Protocol, Src: 192.168.1.101 (192.168.1.101), Dst: 213.186.88.114 (213.186.88.114)
Transmission Control Protocol, Src Port: hhb-gateway (1136), Dst Port: ftp (21), Seq: 22, Ack: 122, Len: 26
File Transfer Protocol (FTP)
PASS ToJestFa\263szyweHas\263o\r\n
Request command: PASS
Request arg: ToJestFa\263szyweHas\263o
0000 00 14 bf 6e 5a 3a 00 03 ff 7c f1 66 08 00 45 00 ...nZ:... .|.f..E.
0010 00 42 34 97 40 00 80 06 d5 e4 c0 a8 01 65 d5 ba .B4.@... .....e..
0020 58 72 04 70 00 15 89 79 d4 7c 9a ae 62 50 18 Xr.p...y .|...bP.
0030 ff 86 31 9d 00 00 50 41 53 53 20 54 6f 4a 65 73 ..1...PA SS ToJes
0040 74 46 61 b3 73 7a 79 77 65 48 61 73 b3 6f 0d 0a tFa.szyw eHas.o..
Ready to load or capture
Packets: 117 Displayed: 117 Marked: 0 Dropped: 0
Profile: Default

APPLICATION LAYER

Threats: Eavesdropping on Transmitted Packets and Password Cracking

THE POP3, IMAP AND SMTP protocols as well as their secure versions (SMTPS, POP3S and IMAPS) cannot provide adequate security against cracking passwords



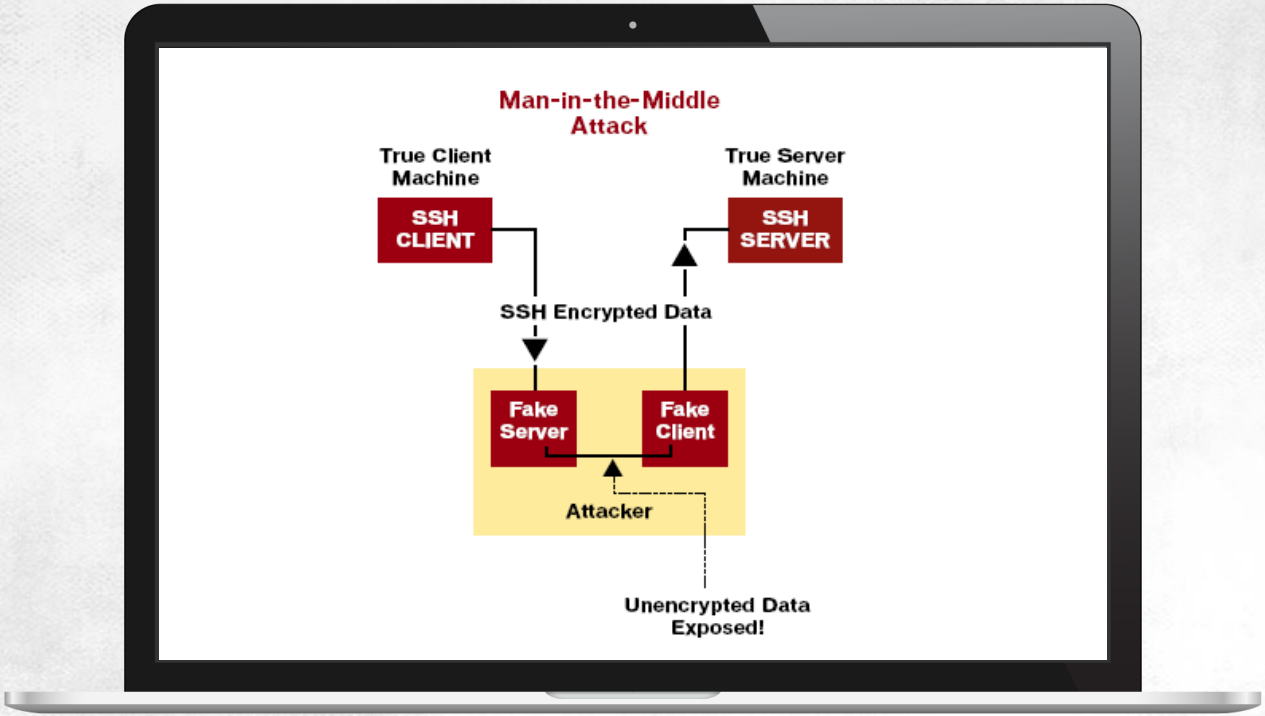
APPLICATION LAYER

Threats: Key Exchange and Man-in-the-Middle-Attack

Host Identification
You are connecting to the host "torni" for the first time.
The host has provided you its identification, a host public key.
The fingerprint of the host public key is:
"xuvin-zitil-ducid-gevil-vysok-buviz-nynun-pinat-tylev-gusez-dyxix"
You can save the host key to the local database by clicking Yes.
You can continue without saving the host key by clicking No.
You can also cancel the connection by clicking Cancel.
Do you want to save the new host key to the local database?
Yes
No
Cancel
Help

APPLICATION LAYER

Threats: Key Exchange and Man-in-the-Middle-Attack



THANKS